IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of: Sunit B. Mangalvedhekar

U.S. Patent Serial No.:   10/085,217

Filing Date: February 27, 2002

Examiner: Frantz Coby

Group No.:   2161

Title: ELECTRONIC FILES PREPARATION FOR

STORAGE IN A SERVER

Mail Stop Appeal Brief - Patents
Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Dear Sir:

CERTIFICATE OF MAILING BY EXPRESS MAIL

I hereby certify that the enclosed Appeal Brief with attached Appendix A (Claims on Appeal), Appendix B (Meske), Appendix C (Siefert), Appendix D, and Appendix E (197 pages), check in the amount of $500.00, Baker Botts return postcard (1 postcard), and this Certificate of Mailing are being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 C.F.R. § 1.10 on this 24th day of August 2005, addressed to the Commissioner for Patents, PO Box 1450, Alexandria, VA 22313-1450.

Willie Jiles

Willie Jiles

Express Mail Receipt
No. EV 733634760 US

**In the United States Patent and Trademark Office
on Appeal from the Examiner to the Board
of Patent Appeals and Interferences**

In re Application of:	Sunit B. Mangalvedhekar
U.S. Patent Serial No.:	10/085,217
Filing Date:	February 27, 2002
Examiner:	Frantz Coby
Group No.:	2161
Title:	ELECTRONIC FILES PREPARATION FOR STORAGE IN A SERVER

MAIL STOP APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

Dear Sir:

Appeal Brief

Appellant has appealed to the Board of Patent Appeals and Interferences ("Board") from the decision of the Examiner mailed March 1, 2005, finally rejecting all pending Claims 1-32. Appellant filed a Notice of Appeal on June 24, 2005, with the statutory fee of $500.00.

Real Party In Interest

This Application is currently owned by UGS PLM Solutions Inc. as indicated by:

an Assignment recorded on February 27, 2002, from the inventor to Electronic Data Systems Corporation, in the Assignment Records of the PTO at Reel 012660, Frame 0069 (3 pages);

an Assignment recorded on February 4, 2004, from Electronic Data Systems Corporation to UGS PLM Solutions Inc., in the Assignment Records of the United States Patent and Trademark Office ("PTO") at Reel 014307, Frame 0325 (7 pages).

Related Appeals and Interferences

To the knowledge of Appellant's counsel, there are no known interferences or judicial proceedings that will directly affect or be directly affected by or have a bearing on the Board's decision regarding this Appeal. With regard to pending Appeals, Appellant filed a Notice of Appeal for Patent Application Serial No. 10/085,218 ("'218 Application") on July 8, 2005. The '218 Application is entitled "ELECTRONIC FILE MANAGEMENT," includes common inventorship to the now appealed Application, is assigned to the same entity as the now appealed Application, and was filed on February 27, 2002. An Appeal Brief for the '218 Application has not been filed at this time. Appellant provides this information for consideration by the Board as the appeal of the '218 Application may potentially be related to, directly affect, be directly affected by, or have bearing on the Board's decision in the now appealed Application.

Status of Claims

Claims 1-32 are pending in this Application, stand rejected pursuant to a final Office Action mailed March 1, 2005 (the "Final Office Action"), and are all presented for appeal. All pending claims are shown in Appendix A, attached hereto, along with an indication of the status of those claims.

Status of Amendments

All amendments submitted by Appellant have been entered by the Examiner.

Summary of Claimed Subject Matter

The present invention relates generally to electronic file preparation for storage in a server. (Page 1, lines 9-11). Specifically, a profile may be generated for a selected file. The profile identifies at least one associated file as associated with the selected file. The selected file, the associated file, and the profile are then transmitted to the server. (Page 3, lines 2-8).

FIGURE 1A is a block diagram of a system 10 that includes a client 14 that is associated with a server 18 by a link 22. Client 14 may be any device that is capable of managing, generating, or storing data, or client 14 may perform other functions related to any data. One example of client 14 is a computer executing suitable client software. Server 18 may be any device that is capable of managing data and that allows at least one client 14 to access data stored in server 18. Link 22 may comprise a medium capable of transporting data between endpoints, such as client 14 and server 18. System 10 may include a plurality of clients 14; however, only one client 14 is shown for clarity of illustration. (Page 6, lines 7-20).

Client 14 includes, in the illustrated embodiment, a processor 32, a memory 28, a storage medium 30, an input device 36, and an output device 40. Processor 32 may be any device operable to process data and execute instructions. An example of processor 32 is the Pentium™ processor available from Intel Corporation; however, other processors may be used. Processor 32 is coupled to link 22. Input device 36, output device 40, memory 28, and storage medium 30 are coupled to processor 32. Memory 28 may be Read Only Memory, Random Access Memory, or may be a removeable medium such as a floppy disk. (Page 6, lines 21-31).

Software program 26 may be any instruction or set of instructions that, when executed by processor 32 of client 14, is operable to transmit, receive, generate, copy, or serve other functions that are related to data. Examples of software program 26 are word processing programs, computer-aided drafting programs such as Solid Edge™ available from Unigraphics Solutions, or other commercial or non-commercial programs. Software program 26 may be a part of an application program such as a drawing package. In the example

shown in FIGURE 1A, software program 26 resides in memory 28, but software program 26 may also reside in storage medium 30. (Page 7, lines 1-12).

Storage medium 30 may be any media that is capable of storing data. An example of storage medium 30 is a conventional hard drive, Compact Disc Read Only memory, Compact Disc Rewritable memory, or other types of electronic data storage. Files 34 reside in storage medium 30 in this embodiment; however, files 34 may also be stored in memory 28. Files 34 may have been generated by client 14 and/or downloaded from server 18. Files 34 may be associated with each other in various ways. Example associations between files 34 are described in conjunction with FIGURE 1B. Storage medium 30 may also store a list 46 describing associations between a given file 34 and its related files, as described in greater detail below. Although only one list 46 is shown, a separate list 46 may be stored in client 14 for each file 34. List 46 may be generated by software program 26. List 46 may alternatively be stored in memory 28. (Page 7, lines 13-29).

Server 18 includes storage medium 52 that stores files 56. Files 56 represent versions of files 34 stored on client 14 that may be accessed by a plurality of clients 56. Files 34 are local versions of files 56 that may be modified and then stored as files 56 on server 18. In one embodiment, files 56 may be managed by a document manager 60. In one embodiment, document manager 60 manages files 56 by maintaining an appropriate file structure, indexing any metadata associated with any of files 56, and accounting for files 56 using identifiers, such as a Uniform Resource Locator ("URL"). Metadata refers to a description of data. In one embodiment, document manager 60 may be a web-based portal, such as Microsoft SharePoint™. However, other types of document managers may be used. (Page 7, line 31 through Page 8, line 14).

FIGURE 1B illustrates an example of the structure of files 34. The illustration of FIGURE 1B may also illustrate an example of the structure of files 56 because files 56 are files 34 that were transferred from client 14. To avoid redundancy of explanation, FIGURE 1B is described using only files 34. (Page 8, lines 14-19).

In one embodiment, files 34 may be assemblies generated by software program 26, which may be a drawing package such as Solid Edge™. In this example, file 34A is designated as a "selected file." A "selected file" refers to one of files 34 that is designated for a data management action, such as being opened, uploaded and/or downloaded. In that sense, any one of files 34 may be a selected file at some point in time. For example, file 34A may be the selected file because file 34A is selected to be downloaded by client 14. (Page 8, lines 20-29).

Selected file 34A may need to use or access one or more of the other files 34. These files that selected file 34A directly uses are referred to herein as "first generation" descendants. For example, the individual part files of a drawing file created by a drawing package such as Solid Edge™ may be categorized into multiple generations of files; the individual part files used directly by the drawing file are first generation descendants. The first generation descendants in this example are files 34B, 34C, and 34D. Each of the first generation descendants, in turn, may directly use additional files. Files used by a first generation descendant file are referred to herein as second generation files. The second generation files in this example are files 34E, 34F, and 34G. File 34B directly uses second generation files 34E and 34F. File 34C directly uses second generation file 34G. File 34D uses no second generation file. A third generation of descendants in this example is represented by files 34H and 34I, both of which are directly used only by file 34G. The generations of descendants may continue depending on the needs of the selected file. (Page 8, line 30 through Page 9, line 20).

Although files 34B through 34I are categorized into multiple generations, all of files 34B through 34I are referred to as associated files of file 34A because files 34B through 34I are descendants of file 34A. A descendant of a selected file is a file that will be used by the selected file or is used by another descendant of the selected file. Files 34B, 34C, and 34D are referred to as immediately associated files of file 34A because file 34A directly uses these files without going through an intermediate file. Once files 34B, 34C, and 34D are selected for access and/or downloading, each of files 34B, 34C, and 34D may be referred to as a selected file. As the selected files, files 34B, 34C, and 34D each may have immediately associated files among the second generation descendants. For example, file 34E and file

34F are immediately associated files of file 34B because from file 34B's point of view, file 34B must access file 34E and file 34F to properly support file 34A. File 34C has the associated files of files 34G, 34H, and 34I, but only file 34G is an immediately associated file because from file 34C's point of view, access to file 34G is necessary to properly support the function of file 34C. File 34D has no immediately associated file. (Page 9, line 21 through Page 10, line 12).

According to the teachings of the invention, an apparatus, a method, and a system are provided that improve the efficiency of using files 34. In one embodiment, efficiency may be improved by generating a profile for each of files 34 that facilitates downloading, all at once, any associated files necessary to use a particular one of files 34. This is advantageous because having all of the files associated with a particular file stored locally in client 14 allows client 14 to work more efficiently with files 34. Furthermore, renamed or relocated files 34 may be located using a profile associated with the renamed or relocated files. (Page 11, lines 10-25).

FIGURE 1C illustrates one embodiment of a profile 38 and a status file 42. A separate profile 38 and status file 42 may be stored for each file 34, in one embodiment. Profile 38 and status file 42 are not explicitly shown in FIGURES 1A and 1B. In one embodiment, profile 38 for any given file 34 may identify files that are immediately associated with the file. For example, for file 34A, profile 38 lists files 34B through 34D as immediately associated files of file 34A. A profile for file 34B (not explicitly shown) may in turn list files 34E and 34F as being immediately associated with file 34B. In another embodiment, profile 38 may identify all of associated files 34B through 34I for file 34A. Files 34 may be identified by profile 38 by any type of identifier, including a URL (as shown in FIGURE 1D) and a globally unique identifier. The globally unique identifier is a unique identifier that is associated with each of files 34 that does not change when the file is renamed or relocated in server 18. Document manager 60, such as Microsoft SharePoint™, may index globally unique identifiers for rapid searching. Other indexable information pertaining to each of files 34 may also be listed in profile 38. In one embodiment, there may be more than one profile 38 for each file 34. For example, one profile 38 of file 34A may identify files 34B through 34D by their respective Uniform Resource Locators, while another

profile of file 34A may identify files 34B through 34D by their respective globally unique identifiers. Listing associated files, immediate or otherwise, in profile 38 facilitates identifying all files used by file 34A, which facilitates downloading those files for use by software program 26. (Page 11, line 26 through Page 12, line 26).

Status file 42 may contain information such as the time of download, check out and check in status, and status of modification of any given file. Each of files 34 may have a status file 42 assigned to it. Status file 42 is generated by software 26, but could be generated by other components, such as document manager 60. Status file 42 may be a cookie file. Having a status file 42 associated with each of files 34 is advantageous because the information pertaining to each of files 34 in status file 42 may be used to facilitate updating files 34 for transferring back to server 18. (Page 12, line 27 through Page 13, line 6).

FIGURE 2 is a flowchart illustrating an embodiment of a method 78 of preparing files for storage in server 18. In one embodiment, method 78 may be implemented by system 10 shown in FIGURE 1. The file structure shown in FIGURE 1B is used as a representative example to describe method 78. Method 78 starts at step 80. At step 84, file 34A is designated as a selected file for transfer to server 18. In one embodiment, file 34A may have been generated by software program 26. Once file 34A has been designated as the selected file, in one embodiment, profile 38 of file 34A identifies files that are immediately associated with file 34A at step 88. Examples of the immediately associated files of file 34A are files 34B through 34D (shown in FIGURE 1B). At step 92, profile 38 for file 34A is generated; in one embodiment, profile 38 lists files immediately associated with file 34A. In one embodiment, other information such as a globally unique identifier for each of the immediately associated files may be listed in profile 38. In another embodiment, profile 38 may identify the immediately associated files using the Uniform Resource Locators. (Page 15, line 13 through Page 16, line 3).

At step 98, software program 26 determines whether any associated files of file 34A is without a profile 38. Steps 84 through 98 are repeated for each of the files 34B through 34I, so that each profile 38 of each associated file identifies that associated file's immediately associated files. For example, file 34B is designated as the selected file at step 84. Then files

34E and 34F are identified as the immediately associated files of file 34B at step 88. At step 92, profile 38 is generated that lists files 34E and 34F as immediately associated files. At step 98, software 26 determines that there are still other associated files requiring generation of a profile listing its descendants. Thus, steps 84 through 98 of method 78 are repeated again. File 34C is designated as the selected file at step 84. Then file 34G is identified as the only immediately associated file of file 34C at step 88. At step 92, a profile 38 is generated that lists file 34G as being the immediately associated file. (Page 16, lines 4-22).

Upon going back to step 84 at step 98 and designating file 34D as the selected file, software 26 recognizes that file 34D has no immediately associated files. As such, in one embodiment, each of the next generation of files are designated as a selected file, and steps 84 through 98 of method 78 are repeated for the remaining associated files until all of the associated files are examined for any immediately associated files. If immediately associated files are found, then the immediately associated files are identified in a profile 38 and associated with the respective file. The end result, in this example, is that a profile 38 of file 34B identifies files 34E and 34F. A profile 38 for file 34C identifies file 34G. A profile 38 of file 34G identifies files 34H and 34I. Each of files 34D, 34H, and 34I has associated with it a profile 38 listing no immediately associated files. (Page 16, line 23 through Page 17, line 8).

Software 26 may identify all associated files of the selected file at step 88, and not just immediately associated files, and generate a profile 38 identifying all associated files, in one embodiment. In that embodiment, steps 84 through 96 are not repeated because all associated files of file 34A are listed in profile 38. (Page 17, lines 9-15).

Then at step 100, file 34A and all of its associated files of file 34B through file 34I are transmitted to server 18 over link 22 for storage as files 56. Method 78 concludes at step 104. Method 78 is advantageous because it allows client 14 to rely on examining the profile 38 for any given file 56 to determine the associated files it uses when downloading that file. Determining the files required by any given file ahead of time allows client 14 to download, all at once, all of the associated files, increasing the efficiency of file access. (Page 17, lines 16-26).

With regard to the independent claims currently under Appeal, Appellant provides the following concise explanation of the subject matter recited in the claim elements. For brevity, <u>Appellant does not necessarily identify every portion of the Specification and drawings relevant to the recited claim elements</u>. Additionally, this explanation should not be used to limit Appellant's claims but is intended to assist the Board in considering the Appeal of this Application.

For example, independent Claim 1 recites the following:

> A method for preparing files for storage in a server (e.g., Figure 2; Page 15, line 13 through Page 17, line 26) comprising:

> generating a profile for a selected file, the profile identifying at least one associated file to be accessed by the selected file (e.g., Figures 1A-1D and 2; Page 11, line 26 through Page 12, line 26; Page 15, line 13- Page 17, line 15); and

> transmitting, to a server, the selected file, the profile, and the at least one associated file (e.g., Figures 1A-1D and 2; Page 7, line 13 through Page 8, line 13; Page 15, lines 3-5; Page 17, lines 16-26).

As another example, independent Claim 9 recites the following:

> A method for file management (e.g., Figures 1A and 3 ; Page 6, lines 10-14; 13 through Page 17, line 26), comprising:

> generating, at a client device, a profile for a selected file that is to be downloaded from a server, the profile identifying all associated files to be accessed by the selected file after the selected file is downloaded from the server (e.g., Figures 1A-1D and 2; Page 10, lines 13-21; Page 11, line 26 through Page 12, line 26; Page 15, line 13- Page 17, line 15);

> transmitting, to a server, the selected file, the profile, and all of the associated files (e.g., Figures 1A-1D and 2; Page 10, lines 13-21; Page 13, line 7 through Page 14, line 3);

> after transmitting the selected file, the profile, and all of the associated files, initiating downloading of the selected file from the server (e.g., Figure 1B; Page 8, lines 20-29; Page 14, lines 4-8; Page 15, lines 3- 5; Page 17, lines 16-26; Page 18, lines 9-12);

identifying all of the associated files by examining the profile (e.g., Figures 1B and 2; Page 14, line 7 through Page 15, line 12; Page 15, line 21 through Page 16, line 3; Page 18, lines 12-14; Page 20, line 14 through Page 21, line 13); and

in response to identifying all of the associated files, initiating downloading of all of the associated files from the server (e.g., FIGURE 4; Page 18, lines 14-26).

As another example, independent Claim 16 recites the following:

A method for preparing a plurality of files for storage in a server (e.g., Figure 2; Page 15, line 13 through Page 17, line 26) comprising:

providing a parent file having at least one level of descendent files (e.g., Figure 1B; Page 8, line 30 through Page 9, line 20);

generating a profile for the parent file identifying all of the descendent files that are immediately associated with the parent file as immediately associated with the parent file (e.g., Figures 1A-1D and 2; Page 11, line 26 through Page 12, line 26; Page 15, line 13- Page 17, line 15);

for each level of the descendent files, generating a profile for each descendent file in the level, the profile identifying all of the descendent files that are immediately associated with the descendent file as immediately associated with the descendent file (e.g., Figures 1A-1D and 2; Page 11, line 26 through Page 12, line 26; Page 15, line 13- Page 17, line 15); and

transmitting the parent file, each descendent file in each level of the descendent files, and the profiles to the server (e.g., Figures 1A-1D and 2; Page 10, lines 13-21; Page 13, line 7 through Page 14, line 3).

As another example, independent Claim 25 recites the following:

An apparatus for preparing files for storage in a server (e.g., Figure 2; Page 15, line 13 through Page 17, line 26) comprising:

software stored on a computer readable medium (e.g., Page 14, line 4 through Page 15, line 12) and operable to:

generate a profile for a selected file, the profile identifying at least one associated file to be accessed by the selected file (e.g., Figures 1A-1D and 2; Page 11, line 26 through Page 12, line 26; Page 15, line 13- Page 17, line 15); and

initiate transmission, to a server, of the selected file, the profile, and the at least one associated file (e.g., Figures 1A-1D and 2; Page 7, line 13 through Page 8, line 13; Page 15, lines 3-5; Page 17, lines 16-26).

Grounds of Rejection to be Reviewed on Appeal

1.   Are Claims 1, 16, and 25 patentable over U.S. Patent No. 5,530,852 issued to Meske et al. ("*Meske*") under 35 U.S.C. § 102(b)?

2.   Are Claims 2-15, 17-24, and 26-32 patentable over the Examiner's proposed combination of *Meske* and U.S. Patent No. 5,721,906 issued to Siefert ("*Siefert*") under 35 U.S.C. § 103(a)?

Grouping of Claims

Appellant has made an effort to group claims to reduce the burden on the Board. In the Argument section of this Appeal Brief, where appropriate, Appellant presents arguments as to why particular claims subject to a ground of rejection are separately patentable from other claims subject to the same ground of rejection. To reduce the number of groups and thereby reduce the burden on the Board, Appellant does not argue individually every claim that recites patentable distinctions over the references cited by the Examiner, particularly in light of the clear allowability of Appellant's independent claims.

The claims of each group provided below may be deemed to stand or fall together for purposes of this Appeal.

With regard to the ground of rejection identified as issue 1 above, the claims subject to that ground of rejection may be grouped together as follows for purposes of this Appeal:

1.   Group 1 may include independent Claims 1, 3, 7, 25, and 28; and

2.   Group 2 may include independent Claim 16, 19, and 23-24.

With regard to the ground of rejection identified as issue 2 above, the claims subject to that ground of rejection may be grouped together as follows for purposes of this Appeal:

1.   Group 3 may include independent Claim 2, 8, 26, and 32;

2.   Group 4 may include Claims 4-6 and 29-31;

3.   Group 5 may include Claims 9-10 and 14-15;

4.   Group 6 may include Claims 11-13;

5.   Group 7 may include Claims 17-18;

6.   Group 8 may include Claims 20-22; and

7.   Group 9 may include Claim 27.

Argument

I.   Issue 1 – The Claims are Patentable over *Meske*

Claims 1, 16, and 25 stand rejected under 35 U.S.C. § 102(b) as being unpatentable over *Meske*. A copy of *Meske* is attached as Appendix B. Appellant respectfully submits that *Meske* fails to support the anticipation rejections of these claims. Appellant respectfully submits that these rejections are therefore improper and should be reversed by the Board.

A.   *Standard*

"A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Verdegaal Bros. v. Union Oil Co. of California*, 2 U.S.P.Q.2d 1051, 1053 (Fed. Cir. 1987); M.P.E.P. § 2131. In addition, "[t]he identical invention must be shown in as complete detail as contained in the . . . claim." M.P.E.P. § 2131 citing *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 U.S.P.Q.2d 1913, 1920 (Fed. Cir. 1989). Furthermore, "[t]he elements must be arranged as required by the claim." *In re Bond*, 910 F.2d 831, 15 U.S.P.Q.2d 1566 (Fed. Cir. 1990); M.P.E.P. § 2131.

B.   *The Meske Reference*

Meske discloses a computer-implemented method and system for retrieving information. (Abstract). According to *Meske*, a user who subscribes to computerized information resources such as the Internet, and various on-line services, such as Compuserve, America Online, Prodigy, and other services manually scans through the various information resources in order to obtain articles, postings, or other files which are of interest. (Column 1, lines 24-28 and 35-39). The user then retrieves articles or files which have subject headings, for example, matching those which the user wishes to read. (Column 1, lines 42-44). The manual scanning process which a user must engage in is very time-consuming. (Column 1, lines 58-59). Thus, the system and method of *Meske* is designed to sort through the large variety of electronic sources in order to generate a subset of the stories available in electronic

form which is tailored to a user's specific interests. (Column 1, line 66 through Column 2, line 3). Specifically, the method and system automatically create hypertext documents from information using profiles and topics, and provide that information to a user. (Column 3, lines 46-49).

To this end, *Meske* discloses a client/server architecture, as illustrated in FIG. 1, wherein user requests 110 for news are sent by a client application program 100 to a server 150 (typically, a remote computer system accessible over the Internet or other communication medium). (Column 3, lines 55-60). Client 100 and server 150 communicate using the functionality provided by Hypertext Transfer Protocol (HTTP). (Column 4, lines 13-14). The server 150 executes the corresponding server software which presents information to the client in the form of HTTP responses. (Column 4, lines 34-36). The HTTP responses correspond with the Web "pages" represented using Hypertext Markup Language (HTML), or other data which is generated by the server. (Column 4, lines 36-39).

A Common Gateway Interlace (CGI) 220 is provided which allows the client program to direct the server to commence execution of a specified program contained within the server. (Column 4, lines 43-46). This may include a search engine which scans received information in the server for presentation to the user controlling the client. (Column 4, lines 46-48). Using this interface, and HTTP, the server may notify the client of the results of that execution upon completion. (Column 4, lines 48-49).

In an implementation, a news source provides an e-mail message at some predetermined time period to the server 210, and the HTML generator 400 parses the message, and creates HTML files which are made available during the client's session. (Column 6, lines 15-19). The e-mail message contains embedded SGML text, which includes profile/topic(key) information. (Column 6, lines 19-20). A profile, in this implementation, is one or many topics. (Column 6, lines 22-23). In another implementation, a profiles/topic may include a USENET newsgroup and subject heading. (Column 6, lines 27-28). Individual topics, in this implementation, are generated via a search of large numbers of publications using heuristic techniques to obtain the topics and group them by profile. (Column 6, lines 29-31).

A separate directory is used for storing each topic. (Column 6, line 37). As illustrated in FIG. 5 and 6b, a directory 501 representing a profile may be created (named "Internet Watch" in the example), if required. (Column 6, lines 37-40). If it already exists, then subdirectories (e.g. 502 and 503, named "Connectivity" and "Making Money") for the topic(s) contained within the profile also are present (and created, if required). (Column 6, lines 40-43). Within each subdirectory, files are created (e.g. 504) which contained the parsed articles themselves contained within the e-mail message 500. (Column 6, lines 43-46). Two types of files are stored for each article: a brief of the article; and the article itself. (Column 6, lines 46-47). Briefs (a.k.a. abstracts) are used by the user during browsing of the results of the information retrieval to determine if a detailed review of the specific article is required. (Column 6, lines 47-50).

The various files created during SGML processing are shown in FIGS. 6a and 6b. (Column 6, lines 51-52). In a specified directory (e.g. the root directory accessible via the Uniform Resource Locator [URL]) the html files index.html 610 and expanded.sub.-- index.html 620 are stored. (Column 6, lines 52-55). The index.html file 610, contains a list of all the profiles which are currently defined (as received in the SGML file). (Column 6, lines 55-57). The expanded.sub.-- index.html file 620, contains a list of the profiles along with their associated anchors referencing a list of abstracts (briefs) for each topic. (Column 6, lines 57-60).

These lists of abstracts are contained in key files (e.g. 620, 630), for each topic. (Column 6, lines 61-62). The index.html file 610, expanded index.html 680, and key files are all created after parsing of the article files, wherein anchors are created in the various files in order to allows hypertext cross-referencing of the various related files and/or documents. (Column 6, lines 62-66). As the SGML file is parsed, profile files (lists of topics) 640, 641,642, etc. . . are created in order to keep track of profiles. (Column 6, line 66 through Column 7, line 1). These are stored in a profiles directory 630. (Column 7, lines 1-2).

As shown in FIG. 6b, a topics directory 650 references each of the topics, stored as directories 660, 661,662, etc. . . (Column 7, lines 3-5). In each topic directory (e.g. 660); a

key file 670 is stored which contains, by topic, references to each of the articles (e.g. 673) contained within the directory. (Column 7, lines 5-7). As will be illustrated below, the key file 670 contained titles represented as anchors to the articles themselves, and associated abstracts (from the brief files--e.g. 672) stored in the directory. (Column 7, lines 7-10). Each brief file, such as 672, also contains an anchor to the article file. Lastly, summary files, such as summary*.html 671, are stored in the directory which contain a previous weeks' summary of titles, represented as anchors, of articles stored in the topic directory. (Column 7, lines 10-15). Summary files are stored with the file specification summary<date#>.html, wherein date# is a julian date for a previous week's date. (Column 7, lines 15-17). Of course, any unique file specification may be used. (Column 7, lines 17-18).

The SGML file is processed twice to obtain relevant information. (Column 7, lines 19-20). First, it is parsed to obtain the articles and briefs for each article. (Column 7, lines 20-21). According to which profiles/topics the articles are relevant to each article and brief, directories, if required are created. (Column 7, lines 21-23). The articles and briefs are then stored in to these subdirectories. (Column 7, lines 23-24). A second pass of the profile and topic subdirectories causes the linkage of the index.html, expanded.sub.-- index.html, key.html, and article html files for each topic for hyperlink cross-referencing. (Column 7, lines 24-27).

C.   Group 1 (Claims 1, 3, 7, 25, and 28)

Claims 1 and 25 stand rejected under 35 U.S.C. § 102(b) as being anticipated by *Meske.*[1] Appellant respectfully submits, however, that the *Meske* reference does not disclose, teach, or suggest each and every element recited in Appellant's Claims 1 and 25.

---

[1] Claims 3, 7, and 28 stand rejected under 35 U.S.C. § 103(a) as being obvious over *Meske* when combined with U.S. Patent No. 5,530,852 issued to Siefert. Although subject to a different grounds of rejection than independent Claims 1 and 25, dependent Claims 3, 7, and 28 have been grouped with Claims 1 and 25 for purposes of this Appeal.

For example, independent Claim 1 recites a method for preparing files for storage in a server that includes:

> generating a profile for a selected file, the profile identifying at least one associated file to be accessed by the selected file; and
> transmitting, to a server, the selected file, the profile, and the at least one associated file.

Appellant respectfully submits that *Meske* does not disclose, teach, or suggest "generating a profile for a selected file, the profile identifying at least one associated file to be accessed by the selected file," as recited in Appellant's Claim 1. As discussed above, the *Meske* system discloses that a news source provides an e-mail message that contains embedded profile/topic(key) information. (Column 6, lines 15-20). A directory 501 representing a profile may be created, and a separate directory is used for storing each topic. (Column 6, line 37-40). Within each subdirectory, files are created (e.g. 504) which contained the parsed articles themselves contained within the e-mail message 500. (Column 6, lines 43-46). Thus, the *Meske* system merely operates to create a directory representing a profile, create subdirectories representing topics within the profile, and store parsed articles within each created subdirectory.

Although the portion of the *Meske* reference that was cited by the Examiner in rejecting Claim 1 describes creating various directories based on a profile and information in the profile, it does not disclose generating a profile for a selected file **where the profile identifies a file to be accessed by the selected file**. In fact, according to *Meske*, a "profile" describes an article, such as a news article, using key words and the topic to which the article belongs. (*See generally*, column 6, lines 19-28). Such a description does not constitute a showing of a "profile" of Claim 1, and no other portion of *Meske* appears to disclose the missing limitation of Claim 1. Simply put, nothing in *Meske* teaches or suggests that its profile identifies files to be accessed by a selected file for which the profile was generated.

In the Final Office Action, the Examiner states that "*Meske* is clear on providing a method for extracting profiles, generating files that contain profiles and topics (associated files) for [use] in accessing data described by the profile (*See Meske* Title)." (Final Office Action, page 8, [sic]). But, even if this were correct, Appellant respectfully submits that

Meske still does not teach generating a profile for a selected file **where the profile identifies a file to be accessed by the selected file**. Stated differently, the profile generated in *Meske* does not identify a file to be accessed by the file for which the profile was generated. To the contrary, the email message for which a profile is generated using the *Meske* system (shown at Figures 4, 5, and 7) does not access a file identified by the profile. Rather, the email message contains the articles and subject headings that are parsed by element 400 of *Meske* (Figure 4). Thus, instead of generating a profile for a selected file that identifies at least one file to be accessed by the selected file and transmitting to a server the selected file, the profile, and the at least one associated file, *Meske* involves receiving an email message containing articles and subject headings for which a more easily searchable directory structure is created (*See Meske,* column 6, lines 44-47 and Figure 7a). In short, *Meske* creates numerous files from the contents of a received email; it does not transfer files to be accessed by the received email — the email accesses no files.

The Examiner also states in the Final Office Action that "[f]urther, to support Meske's intention of accessing data, Meske provides mechanisms for retrieving information containing a list of profiles and corresponding topic for each of the list of profiles." (Final Office Action, page 8 (citing *Meske*, column 2, lines 56-61)). But even if this were correct, *Meske* still does not teach generating a profile for a selected file **wherein the profile identifies a file to be accessed by the selected file**. The cited portion of *Meske* states "[i]n another embodiment a computer-implemented method and apparatus for retrieving information includes using a hypertext transfer protocol to display to a user a display generated from a first markup language, containing a list [sic] a profiles, and at least one corresponding topic for each of the list of profiles." This portion of *Meske* does not involve generating a profile for a selected file wherein the **profile identifies a file to be accessed by the selected file**. Neither the profiles nor the topics referred to in *Meske* are a selected file that accesses a file identified in a profile.

As another example of the deficiencies of *Meske*, Appellant respectfully submits that *Meske* does not disclose, teach, or suggest "transmitting, to a server, the selected file, the profile, and the at least one associated file," as recited in Appellant's Claim 1. The Examiner relies upon Figures 1 and 2 of *Meske* for disclosure of the recited features and operations.

Appellant submits, however, that the mere illustration of a client server in Figures 1 and 2 of *Meske* is not sufficient to establish that *Meske* teaches "transmitting to the server, the selected file, the profile, and the at least one associated file [to be accessed by the selected file]." Rather, Figures 1 and 2 of *Meske* merely show arrows that indicate that the client device and the server are operable to communicate with each other. The description of *Meske* that corresponds with Figures 1 and 2 indicates that the figures illustrate the transmission of a request for information from a client device to a server, and a receipt of a response at the client device. (*See*, column 3, lines 56-60 and column 4, lines 34-39). However, neither the identified set of figures nor the associated description shows transmitting to a server a selected file, a profile that identifies a file to be accessed by the selected file, and the file identified by the profile.

Further, the portions of *Meske* identified in the Final Office Action as showing this limitation in fact describe handling a file of information and generating additional files in response to the receipt of the file. (Column 2, line 20 through Column 3, line 9). It appears that the main point of *Meske* is to take an email that contains numerous articles with associated headings and create a more easily searchable file structure (*See* Background and Figures 4, 5, 6a, 6b, and 7a). Thus, the client server of Figures 1 and 2 is provided to allow a user to retrieve certain files stored on the server that the user deems pertinent. However, *Meske* does not involve the transmission to a server of a selected file, a profile identifying at least one associated file to be accessed by the selected file, and the at least one associated file. Indeed, it appears that the only file transmitted to the server in *Meske* is the email file (*See* Figure 4, for example).

For at least these reasons, Appellant respectfully submits that *Meske* fails to disclose, teach, or suggest each and every limitation recited in Appellant's Claim 1. For at least these reasons, Appellant respectfully submits that the rejection of independent Claim 1 and its dependent claims (including Claims 3 and 7) is improper and should be reversed by the Board. For at least analogous reasons, Appellant respectfully submits that the rejection of independent Claim 25 and its dependent claims (including Claim 28) is improper and should be reversed by the Board.

D.   *Group 2 (Claim 16, 19, and 23-24)*

Claim 16 stands rejected under 35 U.S.C. § 102(b) as being anticipated by *Meske*.[2] Appellant respectfully submits, however, that the *Meske* reference does not disclose, teach, or suggest each and every element recited in Appellant's Claim 16.

Independent Claim 16 recites a method for preparing a plurality of files for storage in a server that includes:

> providing a parent file having at least one level of descendent files;
> generating a profile for the parent file identifying all of the descendent files that are immediately associated with the parent as immediately associated with the parent file;
> for each level of the descendent files, generating a profile for each descendent file in the level, the profile identifying all of the descendent files that are immediately associated with the descendent file as immediately associated with the descendent file; and
> transmitting the parent file, each descendent file in each level of the descendent files, and the profiles to the server.

Accordingly, independent Claim 16 recites certain features and operations that are similar to the features discussed above with respect to independent Claim 1. Thus, for reasons analogous to those discussed above with regard to Claim 1, Appellant respectfully submits that *Meske* does not disclose, teach, or suggest each and every element as set forth in Appellant's Claim 16. Rather, the *Meske* system merely operates to create a directory representing a profile, create subdirectories representing topics within the profile, and store parsed articles within each created subdirectory. For the reasons described above, Appellant submits that the "profile" of *Meske* is not analogous to the "profile" recited in Appellant's Claim 16 and that *Meske* does not involve the transmission to a server of a parent file, a descendent file in each level of the descendent files, and the profiles to the server. Indeed,

---

[2] Claims 19 and 23-24 stand rejected under 35 U.S.C. § 103(a) as being obvious over *Meske* when combined with U.S. Patent No. 5,530,852 issued to Siefert. Although subject to a different grounds of rejection than independent Claim 16, dependent Claims 19 and 23-24 have been grouped with Claim 16 for purposes of this Appeal.

and as described above, it appears that the only file transmitted to the server in *Meske* is the email file (*See* Figure 4, for example).

Additionally, Appellant respectfully submits that Claim 16 is allowable also because *Meske* does not teach or suggest "generating a profile for the parent file identifying **all of the descendent files** that are **immediately associated with** the parent file as immediately associated with the parent file," as recited by Claim 16. In the Office Action, the Examiner asserts that Figures 6A-6B and Column 6, line 52 through Column 7, line 29 disclose the recited features and operations. Appellant respectfully disagrees. In fact, the relied upon figures only generally depict details of files that are created in a server, and the relied upon portion of the description merely mentions the word "profile." Neither of the identified figures nor the portion of the description describe the profile as identifying all of the descendent files that are immediately associated with the parent file.

In the Final Office Action, the Examiner states that "*Meske* discloses generating a profile for a selected file, the profile identifying at least one associated file to be accessed by the selected file. In addition, Meske has shown in Figures 6a-6b that files format include parent file, which are also identified where a profile for the parent is generated." (Final Office Action, page 9). But even if the Examiner's characterization of *Meske* is correct, the Examiner has not established where in *Meske* it is taught that the profile identifies all of the descendent files that are immediately associated with the parent file, as described above.

As a further example of the deficiencies of *Meske*, Appellant respectfully submits that the relied upon reference does not disclose, teach, or suggest "transmitting the parent file, each **descendent file in each level of the descendent files**, and the profiles to the server," as recited in Appellant's Claim 16. Again, for reasons analogous to those discussed above with regard to Claim 1, Appellant submits that neither the identified set of figures (Figures 1 and 2) nor the associated description (Column 3, lines 56-60 and column 4, lines 34-39) show transmitting to a server a parent file, each descendent file in each level of descendent files, and the profiles. Indeed, it appears that the only file transmitted to the server in *Meske* is the email file (*See* Figure 4, for example).

For at least these reasons, Appellant respectfully submits that *Meske* fails to disclose, teach, or suggest each and every limitation recited in Appellant's Claim 16. For at least these reasons, Appellant respectfully submits that the rejection of independent Claim 16 and its dependent claims (including Claims 19 and 23-24) is improper and should be reversed by the Board.

II. Issue 2 – The Claims are Patentable over the Proposed *Meske-Siefert* Combination

Claims 2-15, 17-24, and 26-32 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over the *Meske-Siefert* combination. A copy of *Meske* is attached as Appendix B, and a copy of *Siefert* is attached as Appendix C. Appellant respectfully submits that the Examiner's proposed *Meske-Siefert* combination fails to support the obviousness rejections of these claims. Appellant respectfully submits that these rejections are therefore improper and should be reversed by the Board.

A. Standard

The question raised under 35 U.S.C. § 103 is whether the prior art taken as a whole would suggest the claimed invention taken as a whole to one of ordinary skill in the art at the time of the invention. *See* 35 U.S.C. § 103(a). Accordingly, even if all elements of a claim are disclosed in various prior art references, which is certainly not the case here as discussed below, the claimed invention taken as a whole cannot be said to be obvious without some reason given in the prior art why one of ordinary skill in the art at the time of the invention would have been prompted to modify the teachings of a reference or combine the teachings of multiple references to arrive at the claimed invention.

The M.P.E.P. sets forth the strict legal standard for establishing a *prima facie* case of obviousness based on modification or combination of prior art references. "To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally

available to one of ordinary skill in the art, to modify the reference or combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references where combined) must teach or suggest all the claim limitations." M.P.E.P. § 2142, 2143. The teaching, suggestion or motivation for the modification or combination and the reasonable expectation of success must both be found in the prior art and cannot be based on an Appellant's disclosure. *See Id.* (citations omitted). "Obviousness can only be established by combining or modifying the teachings of the prior art to produce the claimed invention where there is some teaching, suggestion, or motivation to do so found either explicitly or implicitly in the references themselves or in the knowledge generally available to one of ordinary skill in the art" at the time of the invention. M.P.E.P. § 2143.01. Even the fact that references *can* be modified or combined does not render the resultant modification or combination obvious unless the prior art teaches or suggests the desirability of the modification or combination. *See Id.* (citations omitted). Moreover, "To establish *prima facie* obviousness of a claimed invention, all the claim limitations must be taught or suggested by the prior art. All words in a claim must be considered in judging the patentability of that claim against the prior art." M.P.E.P. § 2143.03 (citations omitted).

The governing Federal Circuit case law makes this strict legal standard even more clear.[3] According to the Federal Circuit, "a showing of a suggestion, teaching, or motivation to combine or modify prior art references is an essential component of an obviousness holding." *In re Sang-Su Lee*, 277 F.3d 1338, 1343, 61 U.S.P.Q.2d 1430, 1433 (Fed. Cir. 2002) (quoting *Brown & Williamson Tobacco Corp. v. Philip Morris Inc.*, 229 F.3d 1120, 1124-25, 56 U.S.P.Q.2d 1456, 1459 (Fed. Cir. 2000)). "Evidence of a suggestion, teaching, or motivation . . . may flow from the prior art references themselves, the knowledge of one of ordinary skill in the art, or, in some cases, the nature of the problem to be solved." *In re Dembiczak*, 175 F.3d 994, 999, 50 U.S.P.Q.2d 1614, 1617 (Fed. Cir. 1999). However, the "range of sources available . . . does not diminish the requirement for actual evidence." *Id.* Although a prior art device "may be capable of being modified to run the way the apparatus is claimed, there must be a suggestion or motivation in the reference to do so." *In re Mills*, 916 F.2d at 682, 16 U.S.P.Q.2d at 1432. *See also In re Rouffet*, 149 F.3d 1350, 1357, 47 U.S.P.Q.2d 1453, 1457-58 (Fed. Cir. 1998) (holding a *prima facie* case of obviousness not

---

[3] Note M.P.E.P. 2145 X.C. ("The Federal Circuit has produced a number of decisions overturning obviousness rejections due to a lack of suggestion in the prior art of the desirability of combining references.").

made where the combination of the references taught every element of the claimed invention but did not provide a motivation to combine); *In Re Jones*, 958 F.2d 347, 351, 21 U.S.P.Q.2d 1941, 1944 (Fed. Cir. 1992) ("Conspicuously missing from this record is any evidence, other than the PTO's speculation (if that can be called evidence) that one of ordinary skill in the herbicidal art would have been motivated to make the modification of the prior art salts necessary to arrive at" the claimed invention.). Even a determination that it would have been obvious to one of ordinary skill in the art at the time of the invention to try the proposed modification or combination is not sufficient to establish a *prima facie* case of obviousness. *See In re Fine*, 837 F.2d 1071, 1075, 5 U.S.P.Q.2d 1596, 1599 (Fed. Cir. 1988).

In addition, the M.P.E.P. and the Federal Circuit repeatedly warn against using an Appellant's disclosure as a blueprint to reconstruct the claimed invention. For example, the M.P.E.P. states, "The tendency to resort to 'hindsight' based upon applicant's disclosure is often difficult to avoid due to the very nature of the examination process. However, impermissible hindsight must be avoided and the legal conclusion must be reached on the basis of the facts gleaned from the prior art." M.P.E.P. § 2142. The governing Federal Circuit cases are equally clear. "A critical step in analyzing the patentability of claims pursuant to [35 U.S.C. § 103] is casting the mind back to the time of invention, to consider the thinking of one of ordinary skill in the art, guided only by the prior art references and the then-accepted wisdom in the field. . . . Close adherence to this methodology is especially important in cases where the very ease with which the invention can be understood may prompt one 'to fall victim to the insidious effect of a hindsight syndrome wherein that which only the invention taught is used against its teacher.'" *In re Kotzab*, 217 F.3d 1365, 1369, 55 U.S.P.Q.2d 1313, 1316 (Fed. Cir. 2000) (citations omitted). In *In re Kotzab*, the Federal Circuit noted that to prevent the use of hindsight based on the invention to defeat patentability of the invention, the court requires the Examiner to show a motivation to combine the references that create the case of obviousness. *See id. See also, e.g., Grain Processing Corp. v. American Maize-Products*, 840 F.2d 902, 907, 5 U.S.P.Q.2d 1788, 1792 (Fed. Cir. 1988). Similarly, in *In re Dembiczak*, the Federal Circuit reversed a finding of obviousness by the Board, explaining that the required evidence of such a teaching, suggestion, or motivation is essential to avoid impermissible hindsight reconstruction of an applicant's invention:

Our case law makes clear that the best defense against the subtle but powerful attraction of hind-sight obviousness analysis is *rigorous application of the requirement for a showing of the teaching or motivation to combine prior art references*. Combining prior art references without evidence of such a suggestion, teaching, or motivation simply takes the inventor's disclosure as a blueprint for piecing together the prior art to defeat patentability—the essence of hindsight.

175 F.3d at 999, 50 U.S.P.Q.2d at 1617 (emphasis added) (citations omitted).

B.   The Meske Reference

The *Meske* reference is discussed above in Section I.B of the Discussion portion of this Appeal Brief.

C.   The Siefert Reference

The *Siefert* reference discloses a system for managing resources, which can take the form of (a) computer-compatible information, such as data files and programs, and (b) non-computer-compatible information, such as data contained on microfiche, and (c) physical objects. The invention contains a descriptive profile for each resource, and allows any user to search all profiles for each resource, and to search the profiles according to "fields" (a database term), such as by location of the resources, or by category of the resources. The user can order delivery of a selected resource, and the system causes delivery of the resource to be executed, irrespective of the form (e.g., physical object) of the resource. (Abstract).

Specifically, *Siefert* discloses that a SERVER, which is a computer or equivalent, acts as a REPOSITORY by holding a collection of RESOURCEs for the benefit of microcomputers, labeled PC's. For ease of explanation, the RESOURCEs can be viewed as computer files. However, *Siefert* acknowledges that the RESOURCEs actually include a vastly larger, and more diverse, collection of objects than mere computer files. RESOURCEs include (a) data, (b) information, and (c) knowledge, both as these terms are generally defined, and also as defined by computer scientists. This data, information, and knowledge can take the form of computer-downloadable data, or other forms, such as printed matter.

Each RESOURCE has an associated PROFILE, which contains descriptive information about the RESOURCE. (Column 4, lines 15-28).

The user of a PC uses the PROFILEs to locates RESOURCEs of interest by searching through the PROFILEs. For example, each PROFILE contains a descriptive title. If a user is a manufacturer of golf equipment, and is developing a new golf ball having improved aerodynamic dimples, the user may search the RESOURCEs by looking for phrases such as "golf ball" or "aerodynamic" in combination with "golf ball" within the PROFILEs. The invention will locate the PROFILEs, and thus the RESOURCEs, containing titles which match the search criteria. (Column 4, liens 29-44).

Regarding the Classification of RESOURCEs, *Siefert* discloses that RESOURCEs can be classified as "physical," as physical objects, such as flex diskettes, videotapes, etc., or "soft," as in computer-downloadable RESOURCEs, such as software. When a user orders a "physical" RESOURCE, a message is sent to the custodian of the RESOURCE, requesting delivery. When a user orders a "soft" RESOURCE, the RESOURCE is downloaded to the user directly. (Column 15, lines 31-40).

Regarding the retrieval of RESOURCEs, *Siefert* discloses that if a user wishes to obtain an item, the user highlights this item, using a mouse (or keyboard, or other actuation device, such as a voice sensor). Then, the user actuates the button labeled "RETRIEVE," causing the display to take the appearance shown in Figure 13. The icon bearing the sub-title "CLS Download," located at the bottom of the Figure, indicates that a down-loading operation is taking place. (Column 11, lines 10-23).

D. The Proposed Meske-Siefert Combination Fails to Disclose, Teach, or Suggest Various Limitations Recited in Appellant's Claims

Claims 2-15, 17-24, and 26-32 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over the *Meske-Siefert* combination. Appellant respectfully submits, however, that Claims 2-15, 17-24, and 26-32 are clearly patentable over the proposed *Meske-Siefert*

combination. Appellant respectfully submits that these rejections are, therefore, improper and should be reversed by the Board.

1.   Group 3 (Claims 2, 8, 26, and 32)

Meske, even when considered in combination with *Siefert*, fails to disclose, teach, or suggest various limitations recited in Appellant's dependent Claims 2, 8, 26, and 32. For example, Claim 2 recites a method for file management that includes "after transmitting the selected file, the profile, and the at least one associated file, initiating downloading of the selected file from the server; identifying the at least one associated file by examining the profile, and in response to identifying the at least one associated file by examining the profile, initiating downloading of the at least one associated file from the server." Claim 26 recites certain substantially similar limitations.

In the Final Office Action, the Examiner acknowledges that the recited features and operations are not shown by *Meske*, but argues that column 4, lines 15-45, column 15, lines 31-40, and column 11, lines 10-57 of *Siefert* disclose the recited claim language. Although *Siefert* describes a profile (*see* Column 4, lines 29-31), *Siefert* does not describe the profile as identifying **all of the associated files to be accessed by the selected file**. (See FIGURE 48 of *Siefert*, which is described as showing an example of a profile of *Siefert*.) Furthermore, although *Siefert* generally describes a "RESOURCE" as being downloaded (*see* Column 15, lines 31-40), *Siefert* does not disclose initiating the download of **all of the associated files that are identified by examining a profile of the selected file**. Similarly, although a portion of *Siefert* describes downloading a "RESOURCE," and once the downloading process is complete, searching for a computer program which was used to create the "RESOURCE" to launch the computer program (*see* Column 11, lines 10-57), the identified portion of *Siefert* also does not disclose initiating the download of a selected file, identifying all of the associated files by examining the profile of the selected file, and initiating the download of all of the associated files.

In the Final Office Action, the Examiner states:

> the Examiner disagrees because it is clear that Siefert provides a mechanism for downloading resources (*See* Siefert Col. 4, lines 15-45; Col. 15, lines 31-40; and Col. 11, lines 10-57). Specifically, Siefert discloses that resources are located at geographically diverse sites. The invention contains a descriptive profile for each resource, and allows any user to reach all profiles, and to search the profiles according to "fields" (a database term), such as by location of the resources, or by category of the resources. The user can order delivery of a selected resource, and the system causes delivery of the resource to be executed, irrespective of the form (e.g. physical object) of the resource.

But even if the Examiner is correct, this described teaching of *Siefert* does not disclose "initiating downloading of the selected file from the server; **identifying all of the associated files by examining the profile; and in response to identifying all of the associated files, initiating downloading of all the associated files from the server**." These elements are completely absent from the teachings of *Meske* and *Siefert*.

For at least these reasons, Appellant respectfully submits that the proposed *Meske-Siefert* combination fails to disclose, teach, or suggest each and every limitation recited in Appellant's Claims 2 and 26. For at least these reasons, Appellant respectfully submits that the rejections of Claims 2 and 26 and their respective dependent claims, including Claims 8 and 32, respectively, are improper and should be reversed by the Board.

2. Group 4 (Claims 4-6 and 29-31)

Meske, even when considered in combination with *Siefert*, fails to disclose, teach, or suggest various limitations recited in Appellant's dependent Claims 4-6 and 29-31. For example, Claim 4 recites a method for preparing files for storage in a server that includes "associating a globally unique identifier with each of the files, wherein the profile additionally identifies the at least one associated file by the respective globally unique identifiers." Claim 29 recites certain substantially similar limitations.

In the Final Office Action, the Examiner acknowledges that the recited features and operations are not shown by *Meske*, but argues that Figures 1A-1E of *Siefert* disclose the recited claim language. However, *Siefert* discloses that "FIG. 1A, as well as FIGS. 1B

through 1D, indicate, for ease of explanation, that the PROFILEs are stored in the same server as the RESOURCES." (Column 4, lines 47-49). *Siefert* further discloses that "the preferred method of storage is shown in FIG. 1E," which includes two types of servers, LOCAL SERVERs and REGIONAL SERVERs. (Column 4, lines 49-52). In the preferred embodiment, "[t]he PROFILEs are stored in the REGIONAL SERVERs, and the RESOURCEs are stored in the LOCAL SERVERs." (Column 4, lines 52-54). Thus, Figures 1A-1E of *Siefert* merely illustrate alternative embodiments for the storage of RESOURCEs and PROFILEs. Certainly, Figures 1A-1E do not disclose, teach, or suggest "associating a globally unique identifier with each of the files, wherein the profile additionally identifies the at least one associated file by the respective globally unique identifiers," as recited in Claim 4 and similarly recited in Claim 29. Appellant respectfully submits that these elements are completely absent from the teachings of *Meske* and *Siefert*.

For at least these reasons, Appellant respectfully submits that the proposed *Meske-Siefert* combination fails to disclose, teach, or suggest each and every limitation recited in Appellant's Claims 4 and 29. For at least these reasons, Appellant respectfully submits that the rejections of Claims 4 and 29 and their respective dependent claims, including Claims 5-6 and 30-31, respectively, are improper and should be reversed by the Board.

3. Group 5 (Claims 9-10 and 14-15)

Meske, even when considered in combination with *Siefert*, fails to disclose, teach, or suggest various limitations recited in Appellant's Claims 9-10 and 14-15. For example, Claim 9 recites a method for file management that includes:

> generating, at a client device, a profile for a selected file that is to be downloaded from a server, the profile identifying all associated files to be accessed by the selected file after the selected file is downloaded from the server;
> transmitting, to a server, the selected file, the profile, and all of the associated files;
> after transmitting the selected file, the profile, and all of the associated files, initiating downloading of the selected file from the server;

identifying all of the associated files by examining the
profile; and
in response to identifying all of the associated files,
initiating downloading of all of the associated files from the server.

Thus, independent Claim 9 recites certain features and operations that are similar to the features discussed above with respect to independent Claim 1. As just one example, independent Claim 9 recites "generating, at a client device, a profile for a selected file that is to be downloaded from a server, the profile identifying all associated files to be accessed by the selected file after the selected file is downloaded from the server." As another example, Claim 9 recites "transmitting, to a server, the selected file, the profile, and all of the associated files." In the Final Office Action, the Examiner again relies upon *Meske* for disclosure of the recited features and operations. Thus, for reasons analogous to those discussed above with regard to Claim 1, Appellant respectfully submits that the proposed *Meske-Siefert* combination does not disclose, teach, or suggest each and every element as set forth in Appellant's Claim 9.

Further, Appellant respectfully submits that independent Claim 9 is also allowable because the proposed *Meske-Siefert* combination does not teach or suggest "after transmitting the selected file, the profile, and all of the associated files, initiating downloading of the selected file from the server; identifying all of the associated files by examining the profile; and in response to identifying all of the associated files, initiating downloading of all of the associated files from the server," as recited by Appellant's Claim 9. In the Final Office Action, the Examiner acknowledges that the recited features and operations are not shown by *Meske*, but argues that column 4, lines 15-45, column 15, lines 31-40, and column 11, lines 10-57 of *Siefert* disclose the recited claim language. Although *Siefert* describes a profile (*see* Column 4, lines 29-31), *Siefert* does not describe the profile as identifying **all of the associated files to be accessed by the selected file**. (See FIGURE 48 of *Siefert*, which is described as showing an example of a profile of *Siefert*.) Furthermore, although *Siefert* generally describes a "RESOURCE" as being downloaded (*see* Column 15, lines 31-40), *Siefert* does not disclose initiating the download of **all of the associated files that are identified by examining a profile of the selected file**. Similarly, although a portion of *Siefert* describes downloading a "RESOURCE," and once the downloading process is complete, searching for a computer program which was used to create the "RESOURCE" to

launch the computer program (*see* Column 11, lines 10-57), the identified portion of *Siefert* also does not disclose initiating the download of a selected file, identifying all of the associated files by examining the profile of the selected file, and initiating the download of all of the associated files.

> In the Final Office Action, the Examiner states:
>
> the Examiner disagrees because it is clear that Siefert provides a mechanism for downloading resources (*See* Siefert Col. 4, lines 15-45; Col. 15, lines 31-40; and Col. 11, lines 10-57). Specifically, Siefert discloses that resources are located at geographically diverse sites. The invention contains a descriptive profile for each resource, and allows any user to reach all profiles, and to search the profiles according to "fields" (a database term), such as by location of the resources, or by category of the resources. The user can order delivery of a selected resource, and the system causes delivery of the resource to be executed, irrespective of the form (e.g. physical object) of the resource.

But even if the Examiner is correct, this described teaching of *Siefert* does not disclose "initiating downloading of the selected file from the server; **identifying all of the associated files by examining the profile; and in response to identifying all of the associated files, initiating downloading of all the associated files from the server**." These elements are completely absent from the teachings of *Meske* and *Siefert*.

As an additional example of the deficiencies of the *Meske-Siefert* combination, Appellant respectfully submits that the proposed combination does not disclose teach or suggest "generating, at a client device, a profile for a selected file that is **to be downloaded from a server**, the profile identifying all associated files to be accessed by the selected file after the selected file is **downloaded from the server**," as recited by Appellant's Claim 9. With respect to this limitation, the Examiner specifically relies upon Column 6, line 38 to Column 7, line 29 of *Meske*. Appellant submits, however, that *Meske* does not show this limitation at least because the identified portion of *Meske* describes files that are being created **at the server**, not the client device. (*See* Column 3, lines 25-27, which states "FIGS. 6a and 6b [which are described by the identified portion of *Meske*] show more details of files **which are created in the server**." [emphasis added]). *Siefert* also does not show this limitation.

Furthermore, Appellant has carefully reviewed the First and Final Office Actions and can find no specific assertion that the above-identified limitation is met by the cited references. Appellant has also been unable to locate any response by the Examiner in the Final Office Action to the Appellant's above argument that the cited references do not teach this limitation.

For at least these reasons, Appellant respectfully submits that the proposed *Meske-Siefert* combination fails to disclose, teach, or suggest each and every limitation recited in Appellant's Claim 9. For at least these reasons, Appellant respectfully submits that the rejections of independent Claim 9 and its dependent claims, including Claims 10 and 14-15, are improper and should be reversed by the Board.

4. Group 6 (Claims 11-13)

Meske, even when considered in combination with *Siefert*, fails to disclose, teach, or suggest various limitations recited in Appellant's dependent Claims 11-13. For example, Claim 11 recites a method for file management that includes "associating a globally unique identifier with each of the files, wherein the profile additionally identifies the at least one of the associated files by the respective globally unique identifiers."

In the Final Office Action, the Examiner acknowledges that the recited features and operations are not shown by *Meske*, but argues that Figures 1A-1E of *Siefert* disclose the recited claim language. Accordingly, for reasons that are analogous to those discussed above with regard to Claim 4, Appellant respectfully submits that *Siefert* does not disclose, teach, or suggest the above recited claim language. To the contrary, the relied upon portions of *Siefert* (Figures 1A-1E) merely illustrate alternative embodiments for the storage of RESOURCEs and PROFILEs. The elements recited in Claim 11 are completely absent from the teachings of *Meske* and *Siefert*.

For at least these reasons, Appellant respectfully submits that the proposed *Meske-Siefert* combination fails to disclose, teach, or suggest each and every limitation recited in Appellant's Claim 11. For at least these reasons, Appellant respectfully submits that the

rejections of Claim 11 and its dependent claims, including Claims 12-13, are improper and should be reversed by the Board.

5. Group 7 (Claims 17-18)

Meske, even when considered in combination with *Siefert*, fails to disclose, teach, or suggest various limitations recited in Appellant's dependent Claims 17-18. For example, Claim 17 recites a method for preparing files for storage in a server that includes "after transmitting the parent file, each descendent file in each level of the descendent files, and the profiles, initiating downloading of the parent file from the server; identifying the descendent files in each level of the descendent files by examining the profiles; and in response to identifying the at least one associated file, initiating downloading of all the descendent files in each level of the descendent files from the server."

In the Final Office Action, the Examiner acknowledges that the recited features and operations are not shown by *Meske*, but argues that column 4, lines 15-45, column 15, lines 31-40, and column 11, lines 10-57 of *Siefert* disclose the recited claim language. Accordingly, for reasons that are analogous to those discussed above with regard to Claim 2, Appellant respectfully submits that *Siefert* does not disclose, teach, or suggest the above recited claim language. To the contrary, although a portion of *Siefert* describes downloading a "RESOURCE," and once the downloading process is complete, searching for a computer program which was used to create the "RESOURCE" to launch the computer program (*see* Column 11, lines 10-57), the relied upon portion of *Siefert* does not disclose initiating downloading of the parent file from the server, identifying the descendent files in each level of the descendent files by examining the profiles, and in response to identifying the at least one associated file, initiating downloading of all the descendent files in each level of the descendent files from the server. The elements recited in Claim 17 are completely absent from the teachings of *Meske* and *Siefert*.

For at least these reasons, Appellant respectfully submits that the proposed *Meske-Siefert* combination fails to disclose, teach, or suggest each and every limitation recited in Appellant's Claim 17. For at least these reasons, Appellant respectfully submits that the

rejections of Claim 17 and its dependent claims, including Claim 18, are improper and should be reversed by the Board.

6.   *Group 8 (Claims 20-22)*

Meske, even when considered in combination with *Siefert*, fails to disclose, teach, or suggest various limitations recited in Appellant's dependent Claims 20-22.   For example, Claim 20 recites a method for preparing files for storage in a server that includes "associating a globally unique identifier with each of the plurality of files, wherein the profile additionally identifies each descendent file by the respective globally unique identifiers."

In the Final Office Action, the Examiner acknowledges that the recited features and operations are not shown by *Meske*, but argues that Figures 1A-1E of *Siefert* disclose the recited claim language. Accordingly, for reasons that are analogous to those discussed above with regard to Claim 4, Appellant respectfully submits that *Siefert* does not disclose, teach, or suggest the above recited claim language. To the contrary, the relied upon portions of *Siefert* (Figures 1A-1E) merely illustrate alternative embodiments for the storage of RESOURCEs and PROFILEs. The elements recited in Claim 20 are completely absent from the teachings of *Meske* and *Siefert*.

For at least these reasons, Appellant respectfully submits that the proposed *Meske-Siefert* combination fails to disclose, teach, or suggest each and every limitation recited in Appellant's Claim 20. For at least these reasons, Appellant respectfully submits that the rejections of Claim 20 and its dependent claims, including Claims 21-22, are improper and should be reversed by the Board.

7.   *Group 9 (Claims 27)*

Meske, even when considered in combination with *Siefert*, fails to disclose, teach, or suggest limitations recited in Appellant's dependent Claim 27. In the Final Office Action, the Examiner summarily rejects Claim 27 since "all the limitations of [this claim] have been noted in the rejection of claims [3-8], 9-15, 16, and 25." (Final Office Action, page 7, [sic]).

Appellant notes, however, that Claim 27 includes features that are distinct from the features recited in Claims 3-8, 9-15, 16, and 25. For example, Claim 27 recites a method for preparing files for storage in a server that includes using software that "comprises a drawing package." Appellant has carefully reviewed the Final Office Action and can find no specific assertion that the above-identified limitation is met by the cited references. Furthermore, since *Meske* is related generally to a method and system that allow "information retrieval" from "computerized information resources, such as the Internet, and various on-line services, such as Compuserve, America Online, Prodigy, and other services" (Column 1, lines 14 and 24-28), and since *Siefert* is generally related to information retrieval from computer databases (Column 1, line 46 through Column 2, line 45), Appellant respectfully submits that neither *Meske* nor *Siefert* disclose, teach, or suggest a method for preparing files for storage in a server that includes using software that "comprises a drawing package," as recited in Claim 27.

For at least these reasons, Appellant respectfully submits that the proposed *Meske-Siefert* combination fails to disclose, teach, or suggest each and every limitation recited in Appellant's Claim 27. For at least these reasons, Appellant respectfully submits that the rejection of Claim 27 is improper and should be reversed by the Board.

E. The Proposed Meske-Siefert Combination is Improper as applied to Groups 3-9 (Claims 2-5, 17-24, and 26-32)

With respect to the Examiner's proposed combination of *Meske* with *Siefert*, the Examiner has not shown anything in *Meske*, *Siefert*, or in the knowledge generally available to those of ordinary skill in the art at the time of the invention that would have taught, suggested, or motivated one of ordinary skill in the art at the time of the invention to combine these references in the manner the Examiner proposes. As discussed above, even if all elements of a claim are disclosed in various prior art references, which is certainly not the case here as discussed above, the claimed invention taken as a whole cannot be said to be obvious without some reason given in the prior art why one of ordinary skill in the art at the time of the invention would have been prompted to modify the teachings of a reference or combine the teachings of multiple references to arrive at the claimed invention. To avoid

burdening the Board, Appellant has chosen not to repeat the entirety of Section II.A. here. Appellant trusts that the Board is fully aware of the strict legal standard the Examiner must satisfy. The mere possibility that the teachings of one reference -- *Siefert* -- might improve the teachings of another reference -- *Meske* --, as the Examiner asserts, does not even remotely provide the required teaching, suggestion, or motivation to combine these references.

The Examiner's summary conclusion at page 4 of the Final Office Action that it would have been obvious to a person of ordinary skill in the art at the time of Appellant's invention to modify the client server system of *Meske* to incorporate the method of downloading files as taught by *Siefert* "to permit management of resources" is not supported by any teaching, suggestion, or motivation in *Meske*, *Siefert*, or knowledge generally available to those of ordinary skill in the art at the time of Appellant's invention. (Final Office Action, page 4). To the contrary, the Examiner's conclusory statement is mere speculation and does not provide the suggestion or motivation necessary to make the proposed combination. Since the Examiner has not provided a sufficient teaching, suggestion, or motivation in the prior art, the Examiner's conclusion of obviousness is improper under the M.P.E.P. and governing Federal Circuit case law.

In making the proposed *Meske-Siefert* combination, the Examiner simply relies upon hindsight. Appellant respectfully submits that the M.P.E.P. and governing Federal Circuit case law summarized above clearly prohibit the hindsight reconstruction the Examiner has employed in making these rejections. To reiterate the pronouncement of the Federal Circuit provided in Section II.A. above:

> Our case law makes clear that the best defense against the subtle but powerful attraction of hind-sight obviousness analysis is *rigorous application of the requirement for a showing of the teaching or motivation to combine prior art references*. Combining prior art references without evidence of such a suggestion, teaching, or motivation simply takes the inventor's disclosure as a blueprint for piecing together the prior art to defeat patentability—the essence of hindsight.

175 F.3d at 999, 50 U.S.P.Q.2d at 1617 (emphasis added) (citations omitted). Appellant respectfully submits that the Examiner has employed the type of hindsight reconstruction explicitly forbidden by the M.P.E.P. and Federal Circuit.

For at least these reasons, Appellant respectfully submits that the rejections of the claims within Groups 3-9 (Claims 2-5, 17-24, and 26-32) are improper and should be reversed by the Board.
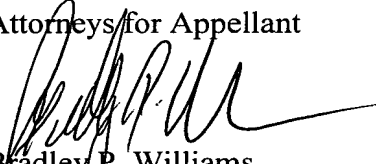
Conclusion

Appellant has demonstrated that the present invention, as claimed, is clearly distinguishable over the prior art cited by the Examiner. Therefore, Appellant respectfully requests the Board to reverse the final rejections and instruct the Examiner to issue a Notice of Allowance with respect to all pending claims.

Appellant encloses a check in the amount of $500 for filing this brief in support of an appeal. Although Appellant believes that no other fees are due, the Commissioner is hereby authorized to charge any fees or credit any overpayment to Deposit Account No. 02-0384 of Baker Botts, L.L.P.

Respectfully submitted,

BAKER BOTTS L.L.P.
Attorneys for Appellant

Bradley P. Williams
Reg. No. 40,227
(214) 953-6447

Date: August 24, 2005

Correspondence Address:

Customer Number: **46629**

Appendix A

IN THE CLAIMS:

1.   (Previously Presented) A method for preparing files for storage in a server comprising:

generating a profile for a selected file, the profile identifying at least one associated file to be accessed by the selected file; and

transmitting, to a server, the selected file, the profile, and the at least one associated file.

2.   (Original) The method of Claim 1, and further comprising:

after transmitting the selected file, the profile, and the at least one associated file, initiating downloading of the selected file from the server;

identifying the at least one associated file by examining the profile; and

in response to identifying the at least one associated file by examining the profile, initiating downloading of the at least one associated file from the server.

3.   (Original) The method of Claim 1, wherein the profile identifies the at least one associated file using a Uniform Resource Locator.

4.   (Original) The method of Claim 1, and further comprising associating a globally unique identifier with each of the files, wherein the profile additionally identifies the at least one associated file by the respective globally unique identifiers.

5.   (Original) The method of Claim 4, and further comprising:

after transmitting the selected file, the profile, and the at least one associated file, determining if any of the at least one associated file is a missing file, wherein the missing file is any of the at least one associated file that has a different identifier than the identifier used by the profile to identify the at least one associated file; and

searching, using a globally unique identifier associated with each of the at least one associated file, for the missing file.

6.   (Original) The method of Claim 5, and further comprising:

determining the different identifier of the missing file; and

updating the profile with the different identifier for the missing file.

7.   (Original) The method of Claim 1, and further comprising:

receiving, at the server, the selected file, the profile, and the at least one

associated file; and

indexing, at the server by a document manager residing in the server, the

profile.

8.   (Original) The method of Claim 2, and further comprising, in response to

initiating downloading of the at least one associated file from the server, storing the at least

one associated file in a memory associated with a client under a local identifier.

9.   (Previously Presented) A method for file management, comprising:

generating, at a client device, a profile for a selected file that is to be

downloaded from a server, the profile identifying all associated files to be accessed by the

selected file after the selected file is downloaded from the server;

transmitting, to a server, the selected file, the profile, and all of the associated

files;

after transmitting the selected file, the profile, and all of the associated files,

initiating downloading of the selected file from the server;

identifying all of the associated files by examining the profile; and

in response to identifying all of the associated files, initiating downloading of

all of the associated files from the server.

10.   (Original) The method of Claim 9, wherein the profile identifies the at least

one associated file using a Uniform Resource Locator.

11.   (Previously Presented) The method of Claim 9, and further comprising

associating a globally unique identifier with each of the files, wherein the profile additionally

identifies at least one of the associated files by the respective globally unique identifiers.

DAL01:866088.1

12.   (Previously Presented) The method of Claim 11, and further comprising:

after transmitting the selected file, the profile, and all of the associated files, determining if any of the associated files is a missing file, wherein the missing file is any of the associated files that has a different identifier than the identifier used by the profile to identify the associated file; and

searching for the missing file using the globally unique identifier of the missing file.

13.   (Original) The method of Claim 12, and further comprising:

determining the different identifier of the missing file; and

updating the profile with the different identifier for the missing file.

14.   (Original) The method of Claim 9, and further comprising:

receiving, at the server, the selected file, the profile, and the at least one associated file; and

indexing, at the server by a document manager residing in the server, the profile.

15.   (Previously Presented) The method of Claim 9, and further comprising, in response to initiating downloading all of the associated files from the server, storing the associated files in a memory associated with a client using a plurality of local identifiers.

16.   (Original) A method for preparing a plurality of files for storage in a server comprising:

providing a parent file having at least one level of descendent files;

generating a profile for the parent file identifying all of the descendent files that are immediately associated with the parent file as immediately associated with the parent file;

for each level of the descendent files, generating a profile for each descendent file in the level, the profile identifying all of the descendent files that are immediately associated with the descendent file as immediately associated with the descendent file; and

transmitting the parent file, each descendent file in each level of the descendent files, and the profiles to the server.

17. (Original) The method of Claim 16, and further comprising:

after transmitting the parent file, each descendent file in each level of the descendent files, and the profiles, initiating downloading of the parent file from the server;

identifying the descendent files in each level of the descendent files by examining the profiles; and

in response to identifying the at least one associated file, initiating downloading of all of the descendent files in each level of the descendent files from the server.

18. (Original) The method of Claim 17, and further comprising compiling a list of all descendent files, and initiating downloading of all of the descendent files identified on the list.

19. (Original) The method of Claim 16, wherein the profile identifies the each of the descendent files in each level of the descendent files using a Uniform Resource Locator.

20. (Original) The method of Claim 16, and further comprising associating a globally unique identifier with each of the plurality of files, wherein the profile additionally identifies each descendent file by the respective globally unique identifiers.

21. (Original) The method of Claim 17, and further comprising:

after transmitting the parent file, the each descendent file in each level of the descendent files, and the profiles to the server, determining if any of the each descendent file is a missing file, wherein the missing file is any of the each descendent file that has a different identifier than the identifier used by the profile to identify the each descendent file; and

searching, using a globally unique identifier associated with the each descendent file, for the missing file.

22.   (Original) The method of Claim 21, and further comprising:

determining the different identifier of the missing file; and

updating the profile with the different identifier for the missing file.

23.   (Original) The method of Claim 16, and further comprising:

receiving, at the server, the parent file, the profiles, and the descendent file;
and

indexing, at the server by a document manager residing in the server, the
profile.

24.   (Original) The method of Claim 17, and further comprising,

in response to initiating downloading of all of the descendent files in each level of the
descendent files from the server, storing the all of the descendent files in each level of the
descendent files in a memory associated with a client under a local identifier.

25.   (Previously Presented) An apparatus for preparing files for storage in a server
comprising:

software stored on a computer readable medium and operable to:

generate a profile for a selected file, the profile identifying at least one
associated file to be accessed by the selected file; and

initiate transmission, to a server, of the selected file, the profile, and the at
least one associated file.

26.   (Original)   The apparatus of Claim 25, wherein the software is further
operable to:

initiate downloading of the selected file from the server;

identify the at least one associated file by examining the profile; and

in response to identifying the at least one associated file, initiate downloading
of the at least one associated file from the server.

27.   (Original)   The apparatus of Claim 25, wherein the software comprises a
drawing package.

DAL01:866088.1

28.   (Original) The apparatus of Claim 25, wherein the profile identifies the at least one associated file using a Uniform Resource Locator.

29.   (Original) The apparatus of Claim 25, wherein the software is further operable to associate a globally unique identifier with each of the files, wherein the profile additionally identifies the at least one associated file by the respective globally unique identifiers.

30.   (Original) The apparatus of Claim 30, wherein the software is further operable to:

after transmitting the selected file, the profile, and the at least one associated file, determine if any of the at least one associated file is a missing file, wherein the missing file is any of the at least one associated file that has a different identifier than the identifier used by the profile to identify the at least one associated file; and

search, using a globally unique identifier associated with each of the at least one associated file, for the missing file.

31.   (Original) The apparatus of Claim 29, wherein the software is further operable to:

determine the different identifier of the missing file; and

update the profile with the different identifier for the missing file.

32.   (Original) The apparatus of Claim 26, wherein the software is further operable to, in response to initiating downloading of the at least one associated file from the server, store the at least one associated file in a memory associated with a client under a local identifier.

Appendix B